



Introdução à Arquitetura Server-side



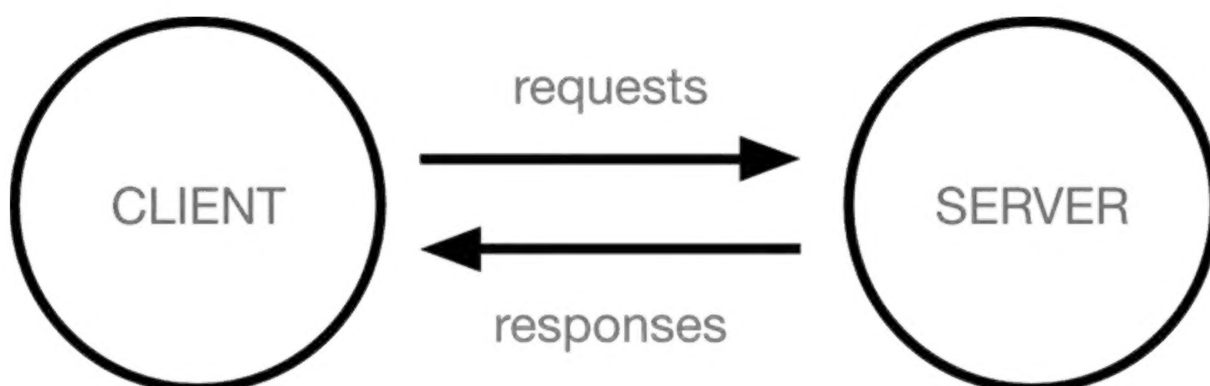

lá, estamos de volta!

Agora que você já conhece a anatomia dos arquivos HTML, como podemos deixar a web mais interativa por meio do CSS e como tirar proveito disso para seus jogos, vamos conversar um pouco sobre como a web funciona.

Será uma visão simplificada do que acontece até que uma página web esteja carregada. No entanto, você vai se beneficiar ao entender o que está acontecendo em segundo plano, pois isso influencia no desempenho de jogos web.

Como a Web Funciona

Para que alguém encontre nossos jogos e páginas web, uma requisição cliente-servidor é realizada com a ajuda de um navegador de internet. A figura a seguir mostra um diagrama simplificado de como isso funciona.



Como é possível notar no diagrama, temos dois tipos de computadores conectados na web: os clientes e os servidores.

- **Clientes** são os dispositivos conectados à internet como computadores e telefones, juntamente com os *softwares* de acesso à web instalados como o navegador, por exemplo, também chamados de **agentes do usuário** (*user agent* no inglês).

- Já os **servidores** são os computadores que armazenam sites e aplicações web na internet. Quando um dispositivo cliente acessa uma página, várias requisições cliente-servidor são feitas.

Isto é, quando você digita um endereço de uma página da internet no seu navegador existe um processo de transferência de dados que dependem de várias tecnologias e infraestruturas de rede. Podemos resumir o processo da seguinte forma:

- Você digita o endereço URL do site. O computador cliente pesquisa no servidor de DNS e encontra o endereço verdadeiro de onde o site está hospedado;

- O navegador manda uma mensagem de requisição HTTP para o servidor, pedindo que envie uma cópia do site. A troca de dados cliente-servidor é feita através da sua rede e conexão à internet usando TCP/IP;

- Após o servidor aprovar a requisição, o servidor envia uma mensagem “200 OK” e então começa a enviar os arquivos do site para o cliente. Isto é feito através da infraestrutura de rede encapsulando os dados em vários pequenos pedaços de informação, chamados pacotes de dados.

- O navegador monta os pequenos pedaços em um site completo e o mostra a você!

A abreviação DNS significa Servidor de Nome de Domínio (*Domain Name Servers*). São servidores na internet que possuem um catálogo de endereços físicos de onde se encontra o servidor onde a página está hospedada. O cliente precisa realizar esta consulta, pois o navegador precisa encontrar onde se encontra a página na internet antes de proceder com a requisição ao servidor.

Os servidores DNS traduzem o endereço URL que digitamos no navegador (como "descomplica.com.br") com o endereço físico real do site (seu endereço IP).

Introdução ao Protocolo HTTP

A abreviação HTTP significa *Hyper Text Transfer Protocol*. Ela trata da comunicação de dados pela internet, pois permite o acesso a recursos como documentos HTML ou bancos de dados, por exemplo, e torna possível a interligação entre os documentos de hipertextos e criação de conteúdos dinâmicos.

Então, a comunicação cliente-servidor na web é feita por meio de requisições e respostas baseadas no HTTP e requisições desse tipo geralmente iniciam a partir de um navegador no computador cliente. Um fluxo HTTP típico de requisições e respostas para um documento HTML se dá da seguinte forma:

- O navegador no computador cliente faz a requisição de uma página HTML disponível na internet. O servidor responde retornando o documento HTML;

- Em seguida, o navegador requisita a folha de estilos anexada no site. O servidor retorna o arquivo CSS;

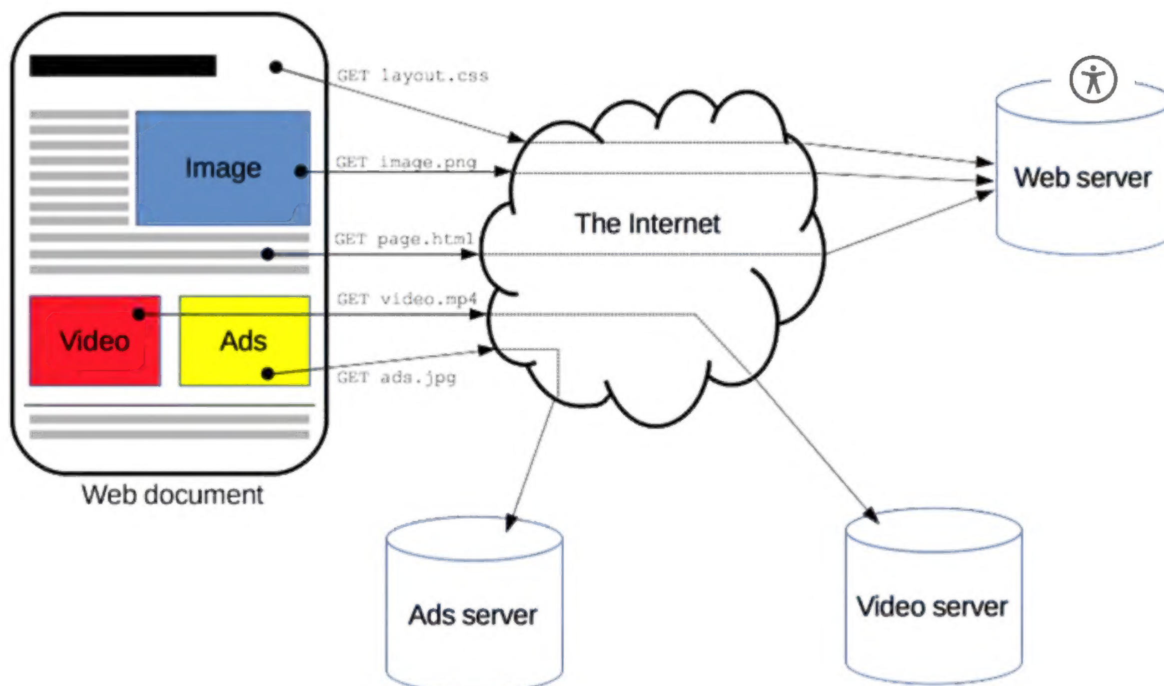
- Se no documento HTML há imagens anexadas, o navegador solicita estes arquivos. Como esperado, o servidor entrega os arquivos ao computador cliente;

- Depois o navegador faz a requisição do arquivo contendo código JavaScript anexados. O servidor responde entregando o arquivo JavaScript.

Como podemos notar, o navegador faz várias requisições HTTP com base nos recursos anexados em nosso documento. Dadas as respostas do servidor, o navegador faz o *download* dos recursos solicitados para depois montar e exibir a página.

Como já tínhamos visto nas aulas anteriores e agora podemos observar com mais detalhes, um documento HTML completo é reconstruído a partir dos diferentes recursos que estão anexados nele como os arquivos de imagens, documentos de folha de estilos CSS, dentre outros.

Veja por exemplo a imagem a seguir, onde temos um documento HTML composto de vários recursos que são buscados de vários servidores disponíveis por meio da internet. Cada recurso desse gera uma requisição HTTP cliente-servidor.



Para ilustrar o processo de download e a recomposição de um documento HTML, observe na próxima imagem uma parte do fluxo HTTP quando acessamos o site da Descomplica. Observe os arquivos de imagem no formato gif e os arquivos CSS, bem como a quantidade de dados e tempo de *download*.

Stat...	Méto...	Domínio	Arquivo	Tipo	Transferido	Taman...	0 ms 2,56 s 5,12 s
200	GET	video-re...	video-react.css	css	12,42 KB	50,45 ...	151 ms
200	GET	fonts.go...	css?family=Source+Sans+Pro:400,70	css	1,74 KB	7,44 KB	221 ms
200	GET	fonts.go...	css?family=Source+Sans+Pro:800,60	css	1,79 KB	9,38 KB	206 ms
200	GET	static.no...	notifia-style.css?v=1	css	11,90 KB	73,96 ...	261 ms
200	GET	fonts.go...	css?family=Open+Sans:600	css	1,54 KB	2,72 KB	74 ms
200	GET	fonts.go...	css?family=Nunito+Sans	css	1,43 KB	1,58 KB	71 ms
200	GET	fonts.go...	css?family=Titan+One	css	1,29 KB	693 B	71 ms
200	GET	fonts.go...	css?family=Montserrat	css	1,43 KB	1,57 KB	79 ms
200	GET	fonts.go...	css?family=Fira+Sans	css	1,47 KB	2,00 KB	77 ms
200	GET	ct.pinter...	/v3/?tid=2613937173680&pd={"em"	gif	532 B	35 B	458 ms
200	GET	t.co	adsct?type=javascript&version=2.0.4	gif	738 B	43 B	656 ms
200	GET	www.go...	/pagead/1p-user-list/827032438/?ra	gif	739 B	42 B	152 ms
200	GET	www.go...	/pagead/1p-user-list/827032438/?ra	gif	739 B	42 B	282 ms
200	GET	www.go...	/pagead/1p-user-list/827032438/?ra	gif	739 B	42 B	147 ms
200	GET	www.go...	/pagead/1p-user-list/827032438/?ra	gif	739 B	42 B	270 ms

Métodos de Requisição HTTP

Os protocolos de rede, definem um conjunto de normas para a comunicação cliente-servidor e isso não é diferente para o HTTP. Assim, no HT stão definidos um conjunto de métodos e algoritmos com o objetivo de expressar qual a intenção para um dado tipo de recurso durante as requisições e na transmissão de dados.

Então, sempre que o navegador iniciar a solicitação ao servidor HTTP, este protocolo precisa que sejam definidos alguns parâmetros:

- **Cabeçalho HTTP (*Headers*):** os cabeçalhos são dados adicionais enviados pelo agente do usuário (cliente) para fornecer mais contexto sobre a transação em andamento entre o cliente e o servidor. Alguns deles ajudarão o servidor a responder da maneira mais adequada.

- **Método HTTP:** o protocolo atualmente contém 8 métodos para fazer uma requisição cliente-servidor: OPTIONS, GET, HEAD, POST, PUT, DELETE, TRACE, CONNECT.

Neste módulo focaremos nos métodos mais comumente utilizados, os métodos **GET** e **POST**.

- GET: quando usamos este método queremos indicar na requisição solicitada que estamos procurando **somente recuperar dados** ou um recurso especificado, sem a possibilidade de modificar algo.

- POST: já neste caso, queremos dizer que vamos **submeter dados** direto a um recurso específico no servidor, geralmente causando alguma mudança no estado da informação armazenada no servidor.

Chegamos ao final do nosso resumo da aula 5. Os conhecimentos apresentados brevemente aqui são essenciais para que você possa projetar seus jogos de forma inteligente, gerenciando os tamanhos de arquivos e

pensando no tempo de resposta. Isso irá contribuir com o polimento de seus jogos na web e sua eficiência.

Em nosso próximo encontro vamos começar nosso percurso na programação com a linguagem JavaScript, vendo detalhes introdutórios para futuramente criarmos interfaces interativas para nossos jogos.

Abraços e até mais!

Atividade Extra

O órgão NIC.br (responsável por coordenar e integrar as iniciativas e serviços da Internet no país) criou uma série contendo 4 vídeos bem descontraídos que ilustram o funcionamento da web. Os vídeos ilustram o que acontece nos bastidores quando você precisa acessar páginas na internet. Assista a estes vídeos para consolidar o conhecimento visto nesta unidade, na seguinte ordem.

- **Parte 1: O protocolo IP**. Disponível em: <https://www.youtube.com/watch?v=HNQD0qJ0TC4>.
- **Parte 2: Sistemas Autônomos, BGP, PTTs.** Disponível em: <https://www.youtube.com/watch?v=C5qNAT_j63M>.
- **Parte 3: DNS.** Disponível em: <https://www.youtube.com/watch?v=ACGuo26Mswl>.
- **Parte 4: Governança da Internet**. Disponível em: <https://www.youtube.com/watch?v=ZYsjMEISR6E>.



Referência Bibliográfica

MDN WEB DOCS. **Introdução à Web.** Disponível em: <https://developer.mozilla.org/pt-BR/docs/Learn/Getting_started_with_the_web>.

PAZ, M. **Webdesign.** Curitiba: InterSaberes, 2021.

W3 Schools. **O que é HTTP?** Disponível em: <https://www.w3schools.com/whatis/whatis_http.asp>.

Ir para exercício